

Anotações Semânticas em Serviços Web para Aprendizagem Colaborativa

Adreia Miranda de Luna, Neide Santos
Universidade do Estado do Rio de Janeiro (UERJ)
Departamento de Informática e Ciência da Computação
andreialuna@superig.com.br, neide@ime.uerj.br

Resumo

Os serviços Web semânticos são um novo paradigma de pesquisa na área de integração de sistemas. Sobre a arquitetura da Web Semântica, espera-se que o pleno potencial de interoperabilidade na Internet seja alcançado. Este trabalho apresenta os serviços Web semânticos e desenvolve um estudo de caso no domínio da aprendizagem colaborativa.

1. Introdução

Nos últimos anos, a *World Wide Web* consolidou-se como rica fonte de informação e veículo usual para a publicação de conteúdo e serviços. Os sistemas de informação baseados na Web movimentam e criam novas oportunidades para os negócios entre as empresas e seus clientes, parceiros e fornecedores, e entre governo e sociedade. Seguindo esta tendência, a educação a distância baseada na Web tem se mostrado uma alternativa educacional imprescindível para democratizar o acesso ao ensino, elevar o padrão de qualidade do processo educativo e incentivar o aprendizado ao longo da vida.

Todavia, este cenário repleto de oportunidades trouxe também uma série de desafios. A integração de sistemas na Internet, por sua vez, demanda o desenvolvimento de padrões abertos de comunicação e de arquitetura, que garantam um grau mínimo de dependência entre as aplicações.

A Internet expandiu o acesso à informação e criou novas oportunidades educacionais. Muitos sistemas voltados para educação, usando a plataforma Web, adotam o enfoque da aprendizagem colaborativa. Sob a perspectiva da colaboração, o conhecimento é uma construção social e o aluno é o elemento central do processo educativo, e o responsável por sua própria aprendizagem. As bases teóricas desta proposta incluem o construtivismo piagetiano, o socioculturalismo e o conceito de ciberespaço. A Aprendizagem Colaborativa Apoiada por Computador (CSCL – *Computer Supported Collaborative Learning*) estuda as formas pelas quais as pessoas podem aprender em grupo com o auxílio do computador. O termo *groupware* designa os aplicativos que implementam os processos de apoio à colaboração e

tipicamente oferecem uma combinação de ferramentas de comunicação, compartilhamento de recursos e coordenação de processos de trabalho [1]. Os ambientes colaborativos de aprendizagem enfatizam as interações sociais, o que implica em que a tecnologia é utilizada para mediar e encorajar ações sociais que constituem a aprendizagem em grupo [2]. A criação destes ambientes requer o emprego das funcionalidades presentes em sistemas de *groupware*.

Sistemas de *groupware* ainda são desenvolvidos como aplicações monolíticas, mas sua concepção está evoluindo para um enfoque centrado em serviços que se comunicam usando a infra-estrutura da Internet, em um cenário em que não importa a localização dos serviços e nem que aplicações são usadas para manipulá-los. Uma arquitetura orientada a serviços é uma escolha adequada quando se considera um ambiente de colaboração cujos usuários trabalham em uma variedade de plataformas – incluindo dispositivos móveis – e cujos recursos não residem necessariamente em servidores centralizados. Os serviços Web aderem de perto aos requisitos desta nova arquitetura, pois é um modelo de desenvolvimento baseado em componentes, com *software* de baixo acoplamento, reutilizáveis, autodescritos, acessíveis pela Internet e neutros quanto à linguagem e à plataforma. A arquitetura não está voltada para um domínio específico de aplicações e o uso destes serviços no suporte de *e-learning* e de *groupware* já foi proposto e implementado.

Um problema, no entanto, pode frear a difusão da arquitetura baseada em serviços Web e ele se relaciona à descoberta – a busca e a seleção – de serviços. Este problema deriva da falta de semântica nas buscas, pois o padrão XML, largamente adotado, uniformiza a representação das mensagens trocadas pelas aplicações, mas carece de expressividade semântica. Os serviços Web semânticos têm suas descrições aumentadas com anotações semânticas para facilitar a automação na descoberta de serviços. Neste sentido, o objetivo deste artigo é discutir os serviços Web semânticos e exemplificar o uso do *framework* WSMO (*Web Service Modeling Ontology*) para imprimir semântica à descrição de um serviço Web que implementa um mecanismo de comunicação entre os participantes de um processo de aprendizagem colaborativa. Com este propósito, a seção 2 discute os serviços Web semânticos; a seção 3 apresenta o uso dos serviços Web no domínio de *e-learning*; a seção 4 introduz a proposta de unir as

tecnologias de serviços Web com as de Web semântica para a busca de serviços no domínio da aprendizagem colaborativa; e, por fim a seção 5 apresenta nossas conclusões.

2. Serviços Web Semânticos

A tecnologia dos serviços Web foi concebida com a promessa de propiciar um novo estágio de interoperabilidade entre as aplicações distribuídas na Internet. Os serviços Web são componentes modulares, autodescritos e autocontidos, acessíveis sobre a Internet [3]. São artefatos de software que oferecem APIs para a interação entre máquinas sobre o protocolo HTTP. A arquitetura dos serviços Web possui três componentes básicos: o protocolo SOAP (*Simple Object Access Protocol*), para a troca de mensagens entre as aplicações; a descrição dos serviços oferecidos, na linguagem WSDL (*Web Services Description Language*); serviços de registro dinâmicos e automatizados, tais como o UDDI (*Universal Description, Discovery, and Integration*), para a localização dos serviços. As metas de interoperabilidade dos serviços Web incluem conectar de forma transparente e automática as aplicações. Os protocolos SOAP, WSDL e UDDI definem como fazer a descoberta e a chamada de métodos em uma aplicação, independente de sua localização ou plataforma. Os problemas de interoperabilidade devem ser resolvidos, então, pelos mecanismos de definição e descoberta desses métodos ou operações. [4].

Um dos princípios fundamentais dos serviços Web está na sua concepção baseada em SOA (*Service-Oriented Architecture*). Em uma arquitetura SOA, além de provedores e clientes dos serviços, há um terceiro componente: o Registro de Serviços (*Service Registry*), uma entidade que funciona como catálogo onde se procura um serviço específico [5]. As especificações SOAP, WSDL e UDDI regem as interações entre clientes, provedores e registros. Os termos que descrevem as interações entre esses elementos são: Publicação (*Publish*), o ato de disponibilizar as informações sobre o serviço em um diretório; Descoberta (*Discovery*), o procedimento de busca e seleção de serviços com base em características funcionais e de qualidade; e Ligação (*Binding*): etapa em que uma aplicação se conecta a um serviço e inicia a interação com ele.

A linguagem para descrição de serviços Web (WSDL) é um vocabulário XML usado para escrever descrições processáveis por máquina. Convém recordar que a linguagem XML surgiu e se estabeleceu como um padrão independente de plataforma para a codificação de mensagens, pavimentando o caminho para a interoperabilidade sintática na Web. O padrão XML uniformiza a representação das mensagens trocadas pelas aplicações, no entanto, carece de expressividade semântica. As descrições dos serviços Web representam um contrato entre clientes e provedores e fornecem uma plataforma para a integração automática de serviços. Os metadados definem os formatos das mensagens, tipos de

dados, protocolos de transporte e formatos de serialização que devem ser utilizados entre o cliente e o provedor do serviço. Também especificam um ou mais endereços onde os provedores do serviço podem ser localizados.

Os aplicativos precisam de informações sobre quais serviços existem e onde encontrá-los. Os mecanismos para a descoberta de serviços variam segundo o grau de automatização. Em um procedimento manual, um ser humano utiliza um serviço de descoberta para encontrar e selecionar uma descrição de serviço que atenda aos seus critérios de funcionalidade ou de outro tipo. Em uma abordagem autônoma, um agente de software realiza esta tarefa, o que pode acontecer em tempo de projeto, ou em mesmo tempo de execução. No caso da descoberta autônoma, a necessidade de descrições semânticas que possam ser compreendidas pelas máquinas cresce consideravelmente.

A chave para o aperfeiçoamento da descoberta de serviços Web é adicionar semântica nas próprias descrições dos serviços e utilizar algoritmos de *matchmaking* semântico. Infelizmente, nem a linguagem WSDL comporta descrições semânticas, nem a especificação UDDI leva em conta a semântica dos serviços Web [6]. Uma visão avançada a respeito da nova geração de serviços Web é o seu desenvolvimento sobre a arquitetura da Web Semântica. Os serviços Web semânticos têm suas descrições aumentadas com anotações semânticas para facilitar maior automação na descoberta, composição, invocação e monitoramento de serviços [7].

A Web Semântica promete tornar o conteúdo da Web legível para as máquinas. Isto viabilizará buscas por conteúdo, não apenas por palavras-chave. A fim de que a Web Semântica funcione, os computadores devem ter acesso a coleções estruturadas de informação. O conceito de ontologia é um dos componentes fundamentais da Web Semântica. Ontologia, neste contexto, é a descrição explícita e precisa de conceitos e relações que existem em certo domínio de conhecimento [8]. Ela requer o uso de um vocabulário específico para descrever os objetos, operações e relações em um determinado domínio e também um conjunto de axiomas lógicos necessários para definir a semântica dos termos. A construção de ontologias e a anotação semântica de recursos têm conferido ímpeto ao desenvolvimento de aplicações na Web Semântica. Essencial para a arquitetura dos serviços Web e suas potencialidades é o fato de que existe, por definição, uma auto-descrição pública dos serviços. No nível mais básico de descrição, devem aparecer detalhes sobre a interface, os métodos disponíveis, as assinaturas dos métodos, os valores de retorno, o endereço de localização do serviço e o protocolo de comunicação que deve ser usado. Nos aspectos relativos à qualidade, a descrição deve conter informações sobre o nível de segurança, tempo de resposta, características de escalabilidade e disponibilidade, e política de tarifação pelo uso. Em um nível mais avançado, a descrição completa de um serviço poderá incluir informações relativas à coreografia e à

orquestração de serviços [5]. Neste contexto, a coreografia define a seqüência e as condições sob as quais múltiplos agentes trocam mensagens, comunicando-se com o objetivo de realizarem alguma função. Orquestração define a seqüência e as condições nas quais um serviço Web invoca outro serviço, com o objetivo de cooperarem para o desempenho de uma tarefa.

O padrão atual para a descrição de serviços Web utiliza a especificação WSDL, que cobre apenas a descrição sintática da interface do serviço. Por isso, as descrições WSDL deixam margem para interpretações ambíguas: dois serviços podem ter descrições semelhantes significando coisas totalmente diferentes; da mesma forma, dois serviços sintaticamente diferentes podem desempenhar a mesma função. Para eliminar a ambigüidade, deve-se definir explicitamente a semântica do serviço Web. Anotações semânticas, no contexto dos serviços Web, são informações adicionais em um documento WSDL que definem o significado de seus elementos pelo mapeamento para um modelo semântico, ou ontologia [9].

Três tipos de elementos explicitamente declarados em uma descrição WSDL podem receber anotações semânticas: operações, mensagens e pré-condições e efeitos. As operações caracterizam as funcionalidades oferecidas pelo serviço; a definição do conteúdo das mensagens trocadas entre clientes e provedores de serviço especifica os parâmetros de entrada e saída das operações e cada operação possui pré-condições e efeitos. As pré-condições são condições lógicas que devem ser avaliadas como verdadeiras antes de se poder executar uma operação. Os efeitos são as conseqüências da execução de uma operação. Quando se relacionam os elementos presentes nas descrições com os conceitos definidos por uma ontologia, o resultado é um entendimento comum a respeito do significado das descrições.

A marcação semântica facilita a automatização das diversas etapas do ciclo de vida dos serviços Web [9]:

- Descoberta de serviços: clientes descrevem seus requisitos funcionais e de interface empregando os termos provenientes de uma ontologia. Técnicas de raciocínio são utilizadas para encontrar a similaridade semântica entre a descrição do serviço e a descrição da requisição;
- Composição de serviços: a descrição da tarefa que se deseja realizar, em termos de seus objetivos, e as anotações dos aspectos funcionais dos serviços existentes orientam a seleção dos serviços que deverão ser combinados e a coordenação desses serviços, isto é, em que ordem eles deverão ser combinados;
- Invocação de serviços: a semântica ajuda a resolver o problema da heterogeneidade de interface, especificando o mapeamento entre diferentes formatos de representação dos dados trocados através de mensagens.

Os sítios da Web devem ser capazes de empregar uma ontologia padrão, consistindo em um conjunto de

classes e propriedades básicas, para declarar e descrever serviços. A proposta de marcação semântica dos serviços Web inclui diversas iniciativas, projetos e linguagens, tais como OWL-S (*OWL-based Web Service Ontology*), WSDL-S (*Web Service Semantics*) e WSMO (*Web Services Modeling Ontology*). OWL-S é uma ontologia de alto nível, que utiliza a linguagem OWL (*Web Ontology Language*) para definir um conjunto de classes e propriedades específicas à descrição de serviços. A classe *Service* está no nível mais alto da ontologia. As propriedades *present*, *describedBy* e *supports* são as propriedades de *Service*, e as classes *ServiceProfile*, *ServiceModel* e *ServiceGrounding* são os respectivos valores dessas propriedades. Cada classe formaliza o conhecimento a respeito dos serviços Web sob uma diferente perspectiva: *ServiceProfile* descreve o que o serviço realiza; *ServiceModel*, como ele funciona e *ServiceGrounding*, como acessá-lo [10].

O projeto WSMO visa pesquisar e desenvolver um completo *framework* para os serviços Web semânticos e consiste em três atividades: WSMO (*Web Services Modeling Ontology*): provê uma especificação formal dos conceitos relacionados aos serviços Web; WSML (*Web Services Modeling Language*): define a linguagem para representar os elementos WSMO; WSMX (*Web Services Modelling eXecution environment*): é a implementação de referência para o *framework* WSMO. Provê um ambiente de execução para os serviços Web. A mais leve das três iniciativas de marcação semântica, WSDL-S é um pequeno conjunto de extensões à linguagem WSDL. A abordagem consiste em aumentar a expressividade dos documentos WSDL empregando conceitos definidos em um modelo semântico externo, mas de forma agnóstica à linguagem de representação do modelo. A idéia de serviços Web já está em uso no domínio da educação.

3. Serviços Web e Serviços de e-Learning

Dentre as propostas de uso de serviços Web nos projetos de e-learning pode-se citar Groove Virtual Office (<http://www.groove.net/>), uma ferramenta proprietária de *groupware*, adota os serviços Web como solução para problemas de interoperabilidade, ou seja, como mecanismo para estender a outras aplicações e plataformas o acesso aos seus dados e componentes, de tal forma que seja possível a usuários de diferentes ferramentas de colaboração compartilhar informações e cooperar em tarefas comuns [11]. O projeto LearnServe (<http://www.learnserve.de/>), da Universidade de Münster, Alemanha, é uma plataforma de *e-learning* construída sobre o paradigma dos serviços Web. O projeto foi concebido a partir da percepção de que um sistema educacional típico é uma coleção de atividades ou processos que interagem com os alunos e com os objetos de aprendizagem. Essa visão permite dividir o sistema em unidades funcionais, que são oferecidas individualmente ou em grupo como serviços Web. Segundo o projeto e a implementação das funcionalidades de *e-learning* na forma de serviços Web

viabilizam sua reutilização, integração e composição. Os alunos teriam a liberdade de adaptarem o conteúdo e a dinâmica dos cursos às suas necessidades e expectativas.

Os projetos do Laboratório de Arquiteturas para Sistemas de Aprendizagem (*Learning Systems Architecture Lab* in <http://www.lsal.cmu.edu/>), da Universidade de Carnegie Mellon, EUA, visam à acessibilidade, estabilidade, interoperabilidade, reutilização e rentabilidade dos sistemas de *e-learning* [12]. O conceito da Arquitetura de Serviços de Aprendizagem (*Learning Services Architecture*) e a Pilha de Serviços de Aprendizagem (*Learning Services Stack*) constituem um *framework* para o desenvolvimento de sistemas educacionais que suportem a interoperabilidade de componentes e conteúdos de aprendizagem.

Xu, Yin e Saddik [13] propõem um *framework* com o objetivo de tornar transparentes as interações entre instituições e provedores de serviços e conteúdos educacionais. Os atores do *framework* são os Provedores de Serviços, que fornecem objetos e conteúdos de aprendizagem, aplicativos de gestão de cursos e de colaboração na forma de serviços Web; Corretores de Serviços, que desempenham as funções de um registro UDDI; e Clientes de Serviços, aplicações que se conectam ao serviço através de mensagens SOAP. Nesta arquitetura, os objetos e as aplicações de aprendizagem estão fracamente acoplados, o que permite que quaisquer dispositivos (*desktops*, *laptops*, *palmtops* ou celulares), em qualquer lugar, utilizem tais componentes, não importando a plataforma em que foram desenvolvidos. Os sistemas baseados nesta plataforma são flexíveis e de fácil integração, pois podem buscar e usar conteúdos e serviços de diferentes fornecedores. A interoperabilidade é garantida pelos padrões de comunicação baseados em XML.

Hill [14] oferece exemplos de serviços Web aplicados à educação, sugerindo que a academia e a indústria já reconheceram os benefícios da tecnologia nos ambientes de aprendizagem. Nota-se, entretanto, que tais iniciativas não têm compromisso explícito, com a semântica envolvida nos serviços Web.

4. Descrição Semântica de Serviços Web em Aprendizagem Colaborativa

O projeto WSMO (*Web Service Modeling Ontology*) propõe um modelo conceitual e uma linguagem formal para descrever semanticamente todos os aspectos dos serviços Web que são relevantes para automatizar a descoberta, composição e invocação desses serviços. WSMO define quatro conceitos principais: as Ontologias (*Ontologies*), que fornecem a terminologia usada pelos outros elementos do modelo; os Serviços Web (*Web Services*), descrições que explicam seus aspectos funcionais e comportamentais (no contexto WSMO, um serviço Web é a sua própria descrição formal); as Metas (*Goals*), que representam os requisitos dos clientes dos serviços e os Mediadores (*Mediators*), descrições semânticas de sistemas responsáveis por resolver os

problemas de heterogeneidade entre ontologias, serviços e metas. WSMO também possui a sua própria família de linguagens: a WSML (*Web Services Modeling Language*), baseada em Lógica de Descrição, Lógica de Primeira Ordem e Programação Lógica [15].

O papel da ontologia é estabelecer um vocabulário comum para os conceitos utilizados ao se descrever as características dos serviços Web. Em WSMO [16], os componentes básicos de uma ontologia são os conceitos, as relações entre esses conceitos, suas instâncias e um conjunto de axiomas, que exploram a semântica dos conceitos e relações. Além disso, WSMO contempla a noção de herança tanto para conceitos quanto para relações. Os conceitos são os componentes básicos de uma ontologia e descrevem elementos que existem no domínio do problema, por meio de um conjunto de atributos nomeados. Os tipos (intervalos) dos atributos podem ser tipos de dados definidos pelo XML Schema, ou outros conceitos. A definição de um conceito é uma expressão lógica que restringe a sua extensão, isto é, o seu conjunto de instâncias. As relações modelam interdependências entre múltiplos conceitos. Uma relação possui um identificador e uma lista ordenada de parâmetros. Uma expressão lógica define o conjunto de instâncias ou tuplas da relação. A criação de instâncias de conceitos ou relações é feita quando se especificam valores concretos para os atributos ou parâmetros.

Em WSMO, as descrições de serviços Web contemplam suas capacidades e interface. A capacidade define o serviço em termos da funcionalidade, e a interface descreve como obter a funcionalidade do serviço, isto é, como se comunicar com ele (coreografia) e como fazê-lo cooperar com outros serviços (orquestração). A capacidade de um serviço compreende quatro conceitos:

- a) Pré-condições (*Preconditions*): condições que precisam ser válidas no espaço de informação antes da execução do serviço;
- b) Suposições (*Assumptions*): condições que precisam ser válidas no mundo real antes da execução do serviço;
- c) Pós-condições (*Postconditions*): descrevem o espaço de informação após a execução do serviço;
- d) Efeitos (*Effects*): descrevem a situação do mundo real após a execução do serviço.

A descoberta é o processo de localização de serviços que preenchem os requisitos do usuário. Realizar uma descoberta envolve fazer o casamento entre a capacidade requisitada pelo usuário e a capacidade oferecida pelo serviço, pela comparação das pré-condições, pós-condições, suposições e efeitos de ambas.

4.1 Exemplo no Domínio da Aprendizagem Colaborativa

A descrição semântica de serviços Web requer o projeto e desenvolvimento de ontologias com foco nos conceitos pertencentes ao domínio de aplicação do serviço e que ajudem a especificar a funcionalidade deste. No domínio da aprendizagem colaborativa, existem diversos esforços para organizar o desenvolvimento e a integração de sistemas sob o paradigma da orientação a serviços (ver

<http://www.elframework.org/>). Um exemplo de aplicação para um ambiente de aprendizagem colaborativa é a possibilidade de professores de curso comunicar eventos aos alunos através de mensagens de texto para celulares (SMS). O SMSWebService é um exemplo de serviço Web responsável pelo envio de mensagens SMS. Os usuários interessados devem possuir conta de acesso, obtida ou renovada mediante a compra de créditos, os quais são consumidos na medida

em que o usuário envia mensagens. O documento SMSWebService.wsdl define a interface do serviço (elemento *Port Type*) e o protocolo por meio do qual essa interface pode ser acessada (elemento *Binding*). O elemento *Port Type* contém as definições das operações disponíveis, enquanto que o elemento *Binding* cria uma ligação HTTP para cada uma dessas operações.

O SMSWebService possui duas operações: *queryAvailableCredit* e *sendSMS*.

a) queryAvailableCredit: consulta o valor monetário dos créditos disponíveis na conta de utilização do serviço. A mensagem de entrada da operação (*queryAvailableCreditRequest*) possui dois parâmetros:

userAccount: o nome da conta de utilização do serviço. *String* alfanumérica sensível ao caso e PIN: o código de acesso à conta de serviço. Valor do tipo inteiro.

A mensagem de saída da operação (*queryAvailableCreditResponse*) possui três parâmetros:

queryAvailableCreditResponse: o código e a descrição do resultado da operação (sucesso ou erro). *String* alfanumérica; *currency*: em caso de sucesso, o código ISO 4217 para a unidade monetária utilizada. *String* com três caracteres alfabéticos maiúsculos; *availableCredit*: em caso de sucesso, o valor dos créditos disponíveis na conta. Valor do tipo ponto flutuante.

b) sendSMS: envia mensagem de texto. A mensagem de entrada da operação (*sendSMSRequest*) possui cinco parâmetros: *userAccount*: o nome da conta de utilização do serviço. *String* alfanumérica sensível ao caso; PIN: o código de acesso à conta de serviço. Valor do tipo inteiro; *phoneNumber*: telefone(s) para o qual(is) a mensagem será enviada. *String* contendo lista de números no formato <Código do País + Código Área da Cidade + N° do Telefone>, separados por vírgula; *message*: texto com a mensagem. *String* alfanumérica com até 160 caracteres; *sendDelay*: retardo no envio da mensagem expresso em minutos. Valor do tipo inteiro.

A mensagem de saída da operação (*sendSMSResponse*) possui um só parâmetro:

sendSMSResponse: o código e a descrição do resultado da operação (sucesso ou erro). *String* alfanumérica.

Elaborando uma descrição semântica: uma descrição baseada na especificação WSDL expressa a sintaxe válida para a invocação do serviço. Para capturar a funcionalidade, WSMO especifica as condições esperadas antes da execução do serviço e os resultados obtidos após a execução. A capacidade do SMSWebService pode ser representada por quatro axiomas lógicos, descritos em linguagem natural:

- Pré-condição: o remetente da mensagem SMS possui conta de acesso ao serviço com crédito disponível;
- Suposição: o(s) destinatário(s) da mensagem possui(em) um telefone celular habilitado para receber mensagens SMS;
- Pós-condição: o crédito disponível na conta do remetente da mensagem é diminuído do valor cobrado pelo serviço;
- Efeito: remetente envia mensagem SMS para o(s) destinatário(s).

As descrições WSMO são documentos escritos na linguagem WSML, utilizando um vocabulário controlado, de significado preciso. Imediatamente percebe-se, por exemplo, que as quatro condições acima empregam as noções de “mensagem”, “remetente”, “destinatário” e “crédito”. É necessário definir de maneira não ambígua a semântica desses termos; por isso foi construída uma ontologia para o serviço de mensagens móveis, com o objetivo de identificar quem faz uso do serviço e quem paga por ele, e o requisito de ser genérica o suficiente para descrever os serviços de mensagens de texto (SMS) ou multimídia (MMS) Para representar todo este universo de discurso, outros termos foram incluídos na ontologia, tais como: “serviço”, “usuário”, “cliente” e “mídia”. Mas, visando não poluir a

ontologia e, ao mesmo tempo, propiciar o reuso das conceituações que independem do domínio específico da comunicação móvel, a criação de uma ontologia monolítica foi substituída pela criação de três ontologias modulares: ontologia para serviços de entrega de mensagens em telefones móveis; ontologia para valores monetários; e, ontologia para mídias digitais.

Foi escolhida a variante WSML-Rule da *Web Service Modeling Language*, como linguagem de formalização das ontologias, e também o editor de ontologias do WSMO Studio, versão 0.5.4, como ferramenta visual de modelagem. O WSMO Studio está disponível como um conjunto de *plug-ins* do Eclipse e agrega uma série de ferramentas para a modelagem de serviços Web semânticos dentro do *framework* WSMO.

Messaging: Ontologia para serviços de entrega de mensagens em telefones móveis

A ontologia para serviços de mensagens é a mais importante para o serviço SMSWebService, pois define o significado dos termos que serão utilizados na descrição das suas pré-condições e efeitos: a Mensagem (*Message*) em si; o remetenteDaMensagem (*messageSender*); a sua Conta (*Account*) de acesso ao serviço; o destinatárioDaMensagem (*messageRecipient*) e o Telefone (*Phone*) para receber as mensagens. Os conceitos desta ontologia fazem referência aos conceitos definidos nas seguintes, utilizando-os como restrições de integridade para os valores dos seus atributos.

Os seguintes fatos estão representados na ontologia de mensagens:

- SMS e MMS são serviços de mensagens
- os serviços de mensagens têm um preço
- o conteúdo de mensagem SMS é somente texto

- d) uma mensagem de texto tem, no máximo, 160 caracteres
- e) o conteúdo de uma mensagem MMS são objetos multimídia
- f) o remetente e os destinatários das mensagens são usuários do serviço;
- g) o cliente do serviço é um usuário que dispõe de uma conta de acesso;
- h) a conta de acesso ao serviço dá direito a certo valor em créditos.
- Alguns conceitos definidos na ontologia de mensagens são apresentados nas tabelas a seguir.

Tabela 1. Conceito Account

Conceito: Account (Conta)			
Descrição: Conta de serviço			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasName</i>	Nome da conta de serviço	<i>String</i>	1..1
<i>hasPIN</i>	Código pessoal de acesso ao serviço	Inteiro	1..1
<i>hasCredit</i>	Crédito disponível para a utilização do serviço	<i>Credit</i>	1..1
<i>ofService</i>	Serviço correspondente à conta	<i>MessageService</i>	0..*

Tabela 2. Conceito Client

Conceito Client (Cliente)			
Descrição: Usuário que paga pelo serviço			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasAccount</i>	Conta de utilização do serviço	<i>Account</i>	0..*
Superconceito			
Identificador	Descrição		
<i>User</i>	Usuário do serviço de mensagens		

Tabela 2. Conceito Message

Conceito: Message (Mensagem)			
Descrição: Mensagem móvel			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>sendTimestamp</i>	Carimbo de data e hora de envio da mensagem	Data e hora	0..1
<i>toUser</i>	Destinatário da mensagem	<i>User</i>	1..*
<i>fromUser</i>	Remetente da mensagem	<i>User</i>	1..1
<i>isDeliveredBy</i>	Serviço utilizado na entrega da mensagem	<i>MessageService</i>	1..1
Subconceitos			
Identificador	Descrição		
MMSmessage	Mensagem multimídia		
SMSmessage	Mensagem de texto		

Tabela 3. Conceito MessageService

Conceito: MessageService (ServiçoDeMensagens)			
Descrição: Serviço de entrega de mensagens em telefones móveis			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasPrice</i>	Preço cobrado pela prestação do serviço	<i>Debt</i>	1..1
<i>hasUsers</i>	Usuários do serviço	<i>User</i>	0..*
Instâncias			
Identificador	Descrição		
MMS	Serviço de mensagens multimídia. Serviço disponível em telefones celulares digitais que permite o envio de mensagens multimídia entre estes equipamentos.		
SMS	Serviço de mensagens curtas. Serviço disponível em telefones celulares digitais que permite o envio de mensagens curtas de texto (até 160 caracteres) entre estes equipamentos.		

Foram também definidas as relações e os axiomas na ontologia de mensagens.

Money: Ontologia para valores monetários

A ontologia para valores monetários fornece à ontologia para serviços de mensagens a definição dos

conceitos Crédito (*Credit*) e Débito (*Debt*). Também define as relações créditoMaiorQueDébito (*creditGreaterThanDebt*) e soma (*sum*) de valores, que serão úteis para a descrição das pré e pós-condições do SMSWebService.

Os seguintes fatos estão representados na ontologia de valores monetários:

- a) Dólar, Euro e Real são unidades monetárias;
- b) crédito é uma quantia não negativa
- c) os preços dos serviços de mensagens são débitos

- d) existe crédito disponível para um pagamento se o seu valor é maior do que a quantia a ser debitada
- Alguns conceitos definidos na ontologia de valores monetários:

Tabela 4. Conceito *MMSmessage*

Conceito <i>MMSmessage</i> (MensagemMMS)			
Descrição: Mensagem multimídia. Definido pelo axioma <i>isMMSmessage</i>			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasMultimediaMessage</i>	O conteúdo da mensagem multimídia	<i>Media</i>	1..*
Superconceito			
Identificador	Descrição		
<i>Message</i>	Mensagem móvel		

Tabela 6. Conceito *Currency*

Conceito: <i>Currency</i> (Moeda)			
Descrição: Código ISO 4217 para unidades monetárias			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasAlphaCode</i>	Código alfabético da unidade monetária (3 letras)	<i>String</i>	1..1
<i>hasNumCode</i>	Código numérico da unidade monetária (3 algarismos)	<i>String</i>	1..1
<i>hasName</i>	Nome da unidade monetária	<i>String</i>	1..1
Instâncias			
Identificador	Descrição		
Euro	EUR – Euro		
Real	BRL – Real Brasileiro		
USDollar	USD – Dólar americano		

Tabela 7. Conceito *Debt*

Conceito: <i>Debt</i> (Débito)			
Descrição: Valor monetário negativo			
Superconceito			
Identificador	Descrição		
<i>Value</i>	Valor monetário.		
Instâncias			
Identificador	Descrição		
<i>MMSprice</i>	Preço do serviço de mensagens multimídia		
<i>SMSprice</i>	Preço do serviço de mensagens de texto		

Tabela 8. Conceito *Value*

Conceito: <i>Value</i> (Valor)			
Descrição: Valor monetário			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasCurrency</i>	Unidade monetária	<i>Currency</i>	1..1
<i>hasAmount</i>	Quantia	Ponto flutuante	1..1
Subconceitos			
Identificador	Descrição		
<i>Credit</i>	Valor monetário não negativo. Definido pelo axioma <i>isCredit</i>		
<i>Debt</i>	Valor monetário negativo. Definido pelo axioma <i>isDebt</i>		

Media: Ontologia para mídias digitais

A ontologia para mídias digitais fornece à ontologia para serviços de mensagens a definição dos conceitos **Mídia** (*Media*) e **Texto** (*Text*).

Os seguintes fatos estão representados na ontologia de mídias digitais:

- a) os tipos de mídia são: texto, imagem, áudio ou vídeo;
- b) A codificação de caracteres de um texto pode seguir o padrão ISO-8859-1.

A seguir, alguns conceitos definidos na ontologia de mídias digitais.

Tabela 9. Conceito *Audio*

Conceito: <i>Audio</i> (Áudio)			
Descrição: Áudio			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasDuration</i>	Tempo de duração do áudio em segundos	Inteiro	0..1
<i>hasSamples</i>	Amostras de áudio	<i>Sample</i>	0..1
<i>hasChannels</i>	Número de canais (fontes de som independentes entre si)	Inteiro	0..1
<i>hasBitRate</i>	Número de bits usados para representar cada segundo do fluxo de áudio (kbps)	Inteiro	0..1
Superconceito			
Identificador	Descrição		
<i>Media</i>	Mídia		

Tabela 10. Conceito *Charset*

Conceito: <i>Charset</i> (ConjuntoDeCaracteres)			
Descrição: Técnica utilizada para reduzir o tamanho de arquivos, para facilitar o transporte e armazenagem			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasName</i>	Identificador IANA de conjuntos de caracteres na Internet	<i>String</i>	1..1
<i>hasScript</i>	Sistema de escrita	<i>String</i>	0..*
Instâncias			
Identificador	Descrição		
ISO-8859-1	Alfabeto Latino 1		
US-ASCII	Padrão ASCII para a codificação de caracteres		
UTF-8	Padrão Unicode para codificação de caracteres de tamanho variável		

Tabela 11. Conceito *CompressionAlgorithm*

Conceito: <i>CompressionAlgorithm</i> (AlgoritmoDeCompressão)			
Descrição: Técnica utilizada para reduzir o tamanho de arquivos, para facilitar o transporte e armazenagem			
Atributos			
Identificador	Descrição	Tipo (Intervalo)	Cardinalidade
<i>hasName</i>	Nome do algoritmo	<i>String</i>	1..1
<i>isLossy</i>	Indica se o método de compressão é com perda de dados	Booleano	1..1

Representação da funcionalidade do serviço utilizando lógica formal

Em WSMO, o elemento serviço Web (*Web Service*) é a descrição formal necessária ao processamento automático dos serviços. Os aspectos funcionais são descritos como a Capacidade do serviço e os aspectos comportamentais, como a sua Interface. Para editar a descrição WSMO do serviço SMSWebService, foi utilizada a WSMO Studio, versão 0.5.4, como ferramenta visual de modelagem, e a variante WSML-Rule da *Web Service Modeling Language*, como linguagem lógica para a construção dos axiomas.

As condições que representam a capacidade do serviço SMSWebService foram descritas em linguagem natural. A descrição formal, no documento SMSWebService.wsml, utiliza o vocabulário fornecido pela ontologia para serviços de mensagens móveis e a linguagem WSML. A capacidade do serviço SMSWebService é definida pelos estados:

a) antes da execução do serviço:

- suposição (*assumption*): a expressão lógica emprega o atributo (*isSMSenabled*) com a semântica de “o telefone do destinatário pode receber mensagens SMS”
- pré-condição (*precondition*): a expressão lógica emprega a relação créditoMaiorQueDébito (*creditGreaterThanDebt*) com a semântica de “o crédito disponível na conta do remetente é maior que o preço do serviço”.

b) após a execução do serviço:

- efeito (*effect*): a expressão lógica emprega os atributos doUsuário (*fromUser*) e paraUsuário (*toUser*) do conceito MensagemSMS (*SMSmessage*) com a semântica de “remetente” e “destinatário”, respectivamente;
- pós-condição (*postcondition*): a expressão lógica emprega o primeiro parâmetro da relação soma (*sum*) com a semântica de “valor do crédito diminuído do preço do serviço”.

WSMO modela a interface do serviço sob duas perspectivas diferentes: a sua Coreografia (a comunicação do cliente com o serviço) e a sua Orquestração (a cooperação com outros serviços).

WSMO permite que mais de uma interface seja definida, permitindo múltiplas formas de interação para o mesmo serviço Web. O formalismo utilizado para a definição de coreografias e orquestrações em WSMO são as Máquinas de Estados Abstratos (ASM – *Abstract State Machines*), também conhecido formalmente como Álgebra Evolutiva. Esses são tópicos avançados, que foram deixados fora do escopo deste trabalho. Como consequência, a interface do SMSWebService não foi especificada na descrição semântica do serviço.

WSMO oferece um modelo conceitual robusto para o desenvolvimento de serviços Web semânticos, que aproveita a filosofia do *Web Service Modeling Framework* (WSMF). Está focado principalmente em resolver a questão da interoperabilidade, pois dedica esforço considerável ao tema da mediação. A noção de mediadores está presente no *framework* para lidar com os problemas de heterogeneidade entre ontologias, serviços Web e metas. Todavia, WSMO é um padrão em refinamento. Alguns aspectos ainda não foram especificados. Por exemplo: como as descrições WSMO devem ser mapeadas na descrição WSDL do serviço ainda é uma questão em aberto, assim como o registro e a consulta das descrições WSMO em diretórios UDDI.

5. Conclusões

A semântica é considerada a abordagem mais completa aos problemas de heterogeneidade da informação. No entanto, o tema suscita duas importantes questões: a criação de linguagens de representação de conhecimento, com uma semântica formal e computável, e a construção de modelos conceituais para uma diversidade de domínios. Os fundamentos da Web Semântica foram lançados com as linguagens XML, RDF e OWL, padrões sobre os quais têm sido desenvolvidas as suas tecnologias e aplicações.

A pesquisa relativa aos serviços Web semânticos se concentra no desenvolvimento de ontologias e *frameworks* para descrição e casamento de serviços Web. Alguns, adotando uma abordagem mais pragmática, se propõem a estender os padrões já existentes, outras iniciativas introduzem padrões complementares, como OWL-S que facilita a descrição de serviços em termos de quatro diferentes ontologias (*Service*, *ServiceProfile*, *ServiceModel* e *ServiceGrounding*) e WSMO separa quatro elementos necessários para a descrição de serviços (Ontologias, Serviços Web, Metas e Mediadores). Uma recente proposta, conhecida como SWSF (*Semantic Web Services Framework*), inclui duas novas especificações: um modelo conceitual para a descrição de serviços – SWSO (*Semantic Web Services Ontology*) – e uma linguagem para especificar o modelo e as instâncias de serviços Web – SWSL (*Semantic Web Services Language*)

A maturidade dos padrões WSDL e UDDI, para publicação e descoberta de serviços, se contrasta com a

diversidade de esforços comprometidos com a realização dos serviços Web semânticos. De fato, novas iniciativas continuam surgindo, cada uma delas preocupada em solucionar uma classe de problemas ignorada pelas demais. Um dos maiores desafios à criação de um modelo de representação semântica abrangente é o dilema da expressividade *versus* complexidade. A capacidade de inferência de um sistema lógico é limitada pelo poder de expressão da linguagem utilizada. Por outro lado, quanto mais expressiva for a lógica, mais complexo será o processo de decisão e mais tempo levará. Em comum, todas as abordagens refletem o entendimento de que as ontologias são essenciais para formalizar o significado das descrições de serviços. Todavia, a validade de uma ontologia depende de esta ser aceita por uma comunidade de usuários. Este acordo em utilizar o vocabulário definido pela ontologia para fazer asserções ou consultas relacionadas ao domínio para o qual a ontologia foi construída é chamado de compromisso ontológico. Portanto, qualquer que seja o *framework* escolhido para modelar a semântica dos serviços, os agentes de *software* só serão efetivamente capazes de interagir dinamicamente com os serviços Web semânticos se houver um esforço prévio, por parte da comunidade de provedores e clientes dos serviços, em torno da criação e do compartilhamento de ontologias para os domínios das aplicações.

Referências Bibliográficas

- [1] Vieira, V. H. Web Services de apoio a aplicações voltadas ao trabalho em grupo. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional). São Carlos, 2006. 128p. USP, ICMC. <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-16052006-191517>>. Acesso em: 6 Fev 2007.
- [2] Santoro, F. M.; Borges, M. R. S.; Santos, N. Um Framework para Estudo de Ambientes de Suporte à Aprendizagem Cooperativa. <<http://www.inf.ufsc.br/sbc-ic/revista/nr4/Sbie98-03-Santoro.htm>>. Acesso em: 3 Abr 2007.
- [3] Cardoso, J; Sheth, A. P. (Eds.) *Semantic Web Services, Process and Applications*. 1ª edição. S.l.: Springer, 2006. 383p.
- [4] Cohen, F. Understanding Web service interoperability. In <<http://www-128.ibm.com/developerworks/webservices/library/ws-inter.html>>. Acesso em: 3 Abr 2007.
- [5] Abinader, J. A.; Lins, R. D. *Web Services em Java*. 1ª edição. Rio de Janeiro: Brasport, 2006. 316p.
- [6] Booth, D. *et al.* (Eds.) *Web Services Architecture*. In <<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>>. Acesso em: 3 Abr 2007.
- [7] Payne, T.; Lassila, O. *Semantic Web Services*. <<http://csdl.computer.org/comp/mags/ex/2004/04/x4014.pdf>>. Acesso em: 29 Out 2006.
- [8] Dias, T. D.; Santos, N. *Web Semântica: Conceitos Básicos e Tecnologias Associadas*. In <<http://magnum.ime.uerj.br/cadernos/cadinf/vol14/7-neide.pdf>>. Acesso em: 17 Dez 2006.

- [9] Akkiraju, R. *et al.* Web Service Semantics - WSDL-S. <<http://www.w3.org/Submission/2005/SUBM-WSDL-S-20051107>>. Acesso em: 3 Abr 2007.
- [10] Martin, D. (Ed.) OWL-S: Semantic Markup for Web Services. <<http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122>>. Acesso em: 3 Abr 2007.
- [11] Dostar, R, S.; Gall, H.; Schmidt, T, R. Web services for Groupware in Distributed and Mobile Collaboration. <http://sirpeople.epfl.ch/rschmidt/papers/Dustdar04WebServices4Groupware.pdf>>. Acesso em: 19 Fev 2007.
- [12] Westerkamp, P. E-Learning as a Web Service. <<http://www.dbs.cs.uni-duesseldorf.de/gvd2004/papers/WesterkampPeter.pdf>>. Acesso em: 4 Abr 2007.
- [13] Xu, Z.; Yin, Z.; Saddik, A. El. A Web Services Oriented Framework for Dynamic E-Learning Systems. <<http://www.mcrlab.uottawa.ca/confs/2003/webservices.pdf>>. Acesso em: 24 Fev 2007.
- [14] Hill, M. A framework to deliver and manage web services-enabled learning objects. Disponível em: <<http://is-alt.massey.ac.nz/reports/157799ResearchFINAL71204.doc>>. Acesso em: 24 Fev 2007.
- [15] Bruijn, J. de; Lausen, H. (Eds.) Web Service Modeling Language (WSML). <<http://www.w3.org/Submission/2005/SUBM-WSML-20050603/>>. Acesso em: 4 Abr 2007.
- [16] Lausen, H.; Polleres, A.; Roman, D.. Web Service Modeling Ontology (WSMO). <<http://www.w3.org/Submission/2005/SUBM-WSMO-20050603/>>. Acesso em: 3 Abr 2007.